

# Crisper-X Server

Generated by Doxygen 1.8.8

Fri Oct 10 2014 20:58:10



# Contents

<b>1</b>	<b>Data Structure Index</b>	<b>1</b>
1.1	Data Structures . . . . .	1
<b>2</b>	<b>File Index</b>	<b>3</b>
2.1	File List . . . . .	3
<b>3</b>	<b>Data Structure Documentation</b>	<b>5</b>
3.1	cJSON Struct Reference . . . . .	5
3.1.1	Detailed Description . . . . .	5
3.2	cJSON_Hooks Struct Reference . . . . .	5
3.2.1	Detailed Description . . . . .	5
3.3	localrow Struct Reference . . . . .	6
3.3.1	Detailed Description . . . . .	6
3.4	node Struct Reference . . . . .	6
3.4.1	Detailed Description . . . . .	6
3.5	restrict Struct Reference . . . . .	6
3.5.1	Detailed Description . . . . .	7
3.6	return_struct Struct Reference . . . . .	7
3.6.1	Detailed Description . . . . .	7
3.7	site Struct Reference . . . . .	7
3.7.1	Detailed Description . . . . .	7
3.8	thread_share_variables Struct Reference . . . . .	8
3.8.1	Detailed Description . . . . .	8
<b>4</b>	<b>File Documentation</b>	<b>9</b>
4.1	gene.cpp File Reference . . . . .	9
4.1.1	Detailed Description . . . . .	9
4.1.2	Function Documentation . . . . .	9
4.1.2.1	get_gene_info . . . . .	9
4.2	localresult.cpp File Reference . . . . .	10
4.2.1	Detailed Description . . . . .	10
4.2.2	Function Documentation . . . . .	10

4.2.2.1	free_mysqlres_local	10
4.2.2.2	localres_count	10
4.2.2.3	make_mysqlres_local	10
4.3	main.cpp File Reference	11
4.3.1	Detailed Description	12
4.3.2	Function Documentation	12
4.3.2.1	main	12
4.4	main.h File Reference	12
4.4.1	Detailed Description	14
4.4.2	Function Documentation	14
4.4.2.1	free_mysqlres_local	14
4.4.2.2	get_Chrr_No	14
4.4.2.3	get_gene_info	14
4.4.2.4	getlineregion	15
4.4.2.5	getRegion	15
4.4.2.6	localres_count	15
4.4.2.7	make_mysqlres_local	15
4.5	region.cpp File Reference	15
4.5.1	Detailed Description	16
4.5.2	Function Documentation	16
4.5.2.1	cmp2	16
4.5.2.2	get_Chrr_No	17
4.5.2.3	getlineregion	17
4.5.2.4	getRegion	17
4.5.2.5	region_wmin	17
4.6	score.cpp File Reference	17
4.6.1	Detailed Description	18
4.6.2	Function Documentation	18
4.6.2.1	cJSON_otj	18
4.6.2.2	score	19
4.6.2.3	subscore	19
4.7	util.cpp File Reference	19
4.7.1	Detailed Description	19
4.8	util.h File Reference	20
4.8.1	Detailed Description	20

# Chapter 1

## Data Structure Index

### 1.1 Data Structures

Here are the data structures with brief descriptions:

<a href="#">cJSON</a>	5
<a href="#">cJSON_Hooks</a>	5
<a href="#">localrow</a>	6
<a href="#">node</a>	
Local structure in getlineregion	6
<a href="#">restrict</a>	6
<a href="#">return_struct</a>	7
<a href="#">site</a>	7
<a href="#">thread_share_variables</a>	
Share data between threads	8



## Chapter 2

# File Index

### 2.1 File List

Here is a list of all documented files with brief descriptions:

<a href="#">gene.cpp</a>	Get gene information . . . . .	9
<a href="#">localresult.cpp</a>	Save data from database to RAM (or hard drive in future version) . . . . .	10
<a href="#">main.cpp</a>	Main function and some assistant functions . . . . .	11
<a href="#">main.h</a>	Head file for whole system . . . . .	12
<a href="#">region.cpp</a>	Get region information . . . . .	15
<a href="#">score.cpp</a>	Include all functions about scoring and counting . . . . .	17
<a href="#">util.cpp</a>	Nothing here now . . . . .	19
<a href="#">util.h</a>	Macros for compiling in both linux and windows . . . . .	20
cJSON/cJSON.c	. . . . .	??
cJSON/cJSON.h	. . . . .	??





## Chapter 3

# Data Structure Documentation

### 3.1 cJSON Struct Reference

#### Data Fields

- struct `cJSON` \* **next**
- struct `cJSON` \* **prev**
- struct `cJSON` \* **child**
- int **type**
- char \* **valuestring**
- int **valueint**
- double **valuedouble**
- char \* **string**

#### 3.1.1 Detailed Description

Definition at line 43 of file cJSON.h.

The documentation for this struct was generated from the following file:

- cJSON/cJSON.h

### 3.2 cJSON\_Hooks Struct Reference

#### Data Fields

- void \*(\* **malloc\_fn** )(size\_t sz)
- void(\* **free\_fn** )(void \*ptr)

#### 3.2.1 Detailed Description

Definition at line 56 of file cJSON.h.

The documentation for this struct was generated from the following file:

- cJSON/cJSON.h

### 3.3 localrow Struct Reference

#### Data Fields

- char **row** [8][LOCALROW\_LEN]
- struct [localrow](#) \* **next**

#### 3.3.1 Detailed Description

Definition at line 64 of file main.h.

The documentation for this struct was generated from the following file:

- [main.h](#)

### 3.4 node Struct Reference

Local structure in getlineregion.

#### Data Fields

- int **s**
- int **t**
- int **type**

#### 3.4.1 Detailed Description

Local structure in getlineregion.

Definition at line 8 of file region.cpp.

The documentation for this struct was generated from the following file:

- [region.cpp](#)

### 3.5 restrict Struct Reference

#### Data Fields

- char **rfc10**
- char **rfc12**
- char **rfc12a**
- char **rfc21**
- char **rfc23**
- char **rfc25**
- char **region** [5+1]
- int **ntlength**

### 3.5.1 Detailed Description

Definition at line 70 of file main.h.

The documentation for this struct was generated from the following file:

- [main.h](#)

## 3.6 return\_struct Struct Reference

### Data Fields

- int **ptts\_num**
- int **len** [NUM\_CHROMOSOME]
- int **num\_chromosome**
- double **dou** [3]

### 3.6.1 Detailed Description

Definition at line 80 of file main.h.

The documentation for this struct was generated from the following file:

- [main.h](#)

## 3.7 site Struct Reference

### Data Fields

- char **nt** [LEN+1]
- char **pam** [PAM\_LEN+1]
- int **index**
- int **count**
- char **chromosome** [100]
- int **region**
- char **strand**
- double **gc**
- double **score**
- double **Sspe\_nor**
- double **Seff\_nor**
- vector< [cJSON](#) \* > **ot**
- [cJSON](#) \* **otj**
- mos\_pthread\_t **ntid**

### 3.7.1 Detailed Description

Definition at line 49 of file main.h.

The documentation for this struct was generated from the following file:

- [main.h](#)

## 3.8 thread\_share\_variables Struct Reference

Share data between threads.

### Data Fields

- [localrow](#) \* **lr**
- [localrow](#) **row**
- int **ini**
- int **type**
- double **r1**

### 3.8.1 Detailed Description

Share data between threads.

See also

[create\\_thread\\_socre](#), [new\\_thread](#), [localrow](#)

Definition at line 154 of file `score.cpp`.

The documentation for this struct was generated from the following file:

- [score.cpp](#)

## Chapter 4

# File Documentation

### 4.1 gene.cpp File Reference

Get gene information.

```
#include "main.h"
```

#### Functions

- int [get\\_gene\\_info](#) (char \*str, const char \*specie\_name, const char \*gene\_name)  
*Get gene information by gene name.*

#### 4.1.1 Detailed Description

Get gene information.

##### Author

Yi Zhao

Definition in file [gene.cpp](#).

#### 4.1.2 Function Documentation

4.1.2.1 int [get\\_gene\\_info](#) ( char \* *str*, const char \* *specie\_name*, const char \* *gene\_name* )

Get gene information by gene name.

##### Parameters

<i>str</i>	Output, the gene's location.
<i>specie_name</i>	Input, the specie's name.
<i>gene_name</i>	Input, the gene's name.

##### Returns

0 for succeed, -1 for failed.

Definition at line 14 of file gene.cpp.

## 4.2 localresult.cpp File Reference

Save data from database to RAM (or hard drive in future version).

```
#include "main.h"
```

### Functions

- int [make\\_mysqlres\\_local](#) ([localrow](#) \*\*localresult, MYSQL\_RES \*result\_t)  
*Save data from MYSQL\_RES to RAM.*
- void [free\\_mysqlres\\_local](#) ([localrow](#) \*localresult)  
*Free data on RAM.*
- int [localres\\_count](#) ([localrow](#) \*lr)  
*Get number of saved sgRNA-Info on RAM.*

### 4.2.1 Detailed Description

Save data from database to RAM (or hard drive in future version).

Author

Yi Zhao

Definition in file [localresult.cpp](#).

### 4.2.2 Function Documentation

#### 4.2.2.1 void free\_mysqlres\_local ( localrow \* localresult )

Free data on RAM.

See also

[localrow](#)

Definition at line 36 of file localresult.cpp.

#### 4.2.2.2 int localres\_count ( localrow \* lr )

Get number of saved sgRNA-Info on RAM.

See also

[localrow](#)

Definition at line 47 of file localresult.cpp.

#### 4.2.2.3 int make\_mysqlres\_local ( localrow \*\* localresult, MYSQL\_RES \* result\_t )

Save data from MYSQL\_RES to RAM.

See also

[localrow](#)

#### Remarks

Data saved in linked list.

Usage:

```
MYSQL_RES *result=mysql_store_result(mysql_conn);
```

```
localrow *localresult;
```

```
int res=make_mysqlres_local(&localresult,result);
```

Definition at line 16 of file localresult.cpp.

## 4.3 main.cpp File Reference

Main function and some assistant functions.

```
#include "main.h"
```

#### Functions

- bool **cmp\_in\_site** ([site](#) a, [site](#) b)
- bool **cmp\_by\_index** ([site](#) a, [site](#) b)
- int **readLine** (FILE \*file)
- [cJSON](#) \* **Create\_array\_of\_anything** ([cJSON](#) \*\*objects, int num)
- int **check\_region** (int i)
- int **check\_rfc** (int i)
- int **check\_pam** (const char \*str, const char \*pam)
- char **dna\_rev\_char** (char ch)
- char \* **dna\_rev** (char \*sr, const char \*s, int len)
- char \* **NomoreSpace** (char \*str)
- char \* **\_NomoreSpace** (char \*str)
- int **check\_req** ([cJSON](#) \*request)
- void **onError** (const char \*msg)
- int [main](#) (int args, char \*argv[])

*Main function.*

#### Variables

- [restrict req\\_restrict](#)  
*brief. hehe*
- int **ini**
- [site](#) **in\_site** [NODE\_SIZE]
- MYSQL \* **my\_conn**
- char **argv\_default** [] ={"specie\":"Saccharomyces-cerevisiae\","length\":"17\","gene\":"AAD15\","pam\":"↵\":"NGG\","rfc\":"100010\}"
- const char \* **region\_info** [] ={"","EXON","INTRON","UTR","INTERGENIC"}
- [localrow](#) \* **localresult**
- mos\_thread\_mutex\_t **mutex**
- mos\_thread\_mutex\_t **mutex\_mysql\_conn**
- mos\_sem\_t **sem\_thread**
- mos\_sem\_t **sem\_thread\_usage**

### 4.3.1 Detailed Description

Main function and some assistant functions.

Author

Yi Zhao

Definition in file [main.cpp](#).

### 4.3.2 Function Documentation

#### 4.3.2.1 `int main ( int args, char * argv[] )`

Main function.

Include Input, output and Database create connect.

Definition at line 183 of file `main.cpp`.

## 4.4 `main.h` File Reference

Head file for whole system.

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
#include <algorithm>
#include <vector>
#include <map>
#include "cJSON/cJSON.h"
#include "mysql.h"
#include "util.h"
```

### Data Structures

- struct [site](#)
- struct [localrow](#)
- struct [restrict](#)
- struct [return\\_struct](#)

### Macros

- `#define PTT_SARS 0`
- `#define PTT_ECOLI 1`
- `#define PTT_SACCHAROMYCETES 2`
- `#define REGION_EXON 1`
- `#define REGION_INTRON 2`
- `#define REGION_UTR 3`
- `#define REGION_INTERGENIC 4`
- `#define REGION_GENE 5`
- `#define LEN 20`
- `#define PAM_LEN 20`



- `#define NUM_NO 4`
- `#define NUM_CHROMOSOME 30`
- `#define GENE_LEN 8000000`
- `#define DCFILE_LEN 8000000`
- `#define NODE_SIZE 1000000`
- `#define LOCALROW_LEN 120`
- `#define MAX_SEM_THREAD 80`
- `#define RETUEN_ERROR -1`
- `#define RETURN_SUCCEED 0`
- `#define MYSQL_CONF_HOST "127.0.0.1"`
- `#define MYSQL_CONF_USERNAME "root"`
- `#define MYSQL_CONF_DB "CasDB"`
- `#define MYSQL_CONF_PASSWD ""`

## Typedefs

- typedef struct [site](#) **site**
- typedef struct [localrow](#) **localrow**
- typedef struct [restrict](#) **restrict**

## Functions

- int **readLine** (FILE \*)
- [cJSON](#) \* **Create\_array\_of\_anything** ([cJSON](#) \*\*objects, int num)
- void **create\_thread\_socre** ([localrow](#) \*lr, [localrow](#) row, int ini, int type, double r1)  
*Function to create new thread.*
- char \* **NomoreSpace** (char \*str)
- char \* **\_NomoreSpace** (char \*str)
- int **get\_gene\_info** (char \*, const char \*, const char \*)  
*Get gene information by gene name.*
- int **get\_Chr\_No** (const char \*, const char \*)  
*Get chromosome's Number(Chr\_No) by specie's name and chromosome's name.*
- [cJSON](#) \* **getlineregion** (int, int, int)  
*Get region information for part of a chromosome.*
- int **getRegion** (int sgrna\_ID, int Chr\_No, int sgrna\_start, int sgrna\_end)  
*Get region information of an sgRNA.*
- int **make\_mysqlres\_local** ([localrow](#) \*\*localresult, MYSQL\_RES \*result\_t)  
*Save data from MYSQL\_RES to RAM.*
- void **free\_mysqlres\_local** ([localrow](#) \*localresult)  
*Free data on RAM.*
- int **localres\_count** ([localrow](#) \*lr)  
*Get number of saved sgRNA-Info on RAM.*

## Variables

- `mos_pthread_mutex_t` **mutex**
- `mos_pthread_mutex_t` **mutex\_mysql\_conn**
- `mos_sem_t` **sem\_thread**
- `mos_sem_t` **sem\_thread\_usage**
- [restrict](#) **req\_restrict**  
*brief. hehe*
- int **ini**

- [site](#) [in\\_site](#) [NODE\_SIZE]
- const char \* [region\\_info](#) []
- [cJSON](#) \* [dc\\_root](#)
- MYSQL \* [my\\_conn](#)

#### 4.4.1 Detailed Description

Head file for whole system.

Author

Yi Zhao

Note

Include all necessary files, most definitions and structure definitions, as well as global variables.

Definition in file [main.h](#).

#### 4.4.2 Function Documentation

##### 4.4.2.1 void free\_mysqlres\_local ( localrow \* localresult )

Free data on RAM.

See also

[localrow](#)

Definition at line 36 of file localresult.cpp.

##### 4.4.2.2 int get\_Chrr\_No ( const char \* specie, const char \* chr\_name )

Get chromosome's Number(Chr\_No) by specie's name and chromosome's name.

Returns

-1 means failed. Otherwise return Chr\_No.

Definition at line 26 of file region.cpp.

##### 4.4.2.3 int get\_gene\_info ( char \* str, const char \* specie\_name, const char \* gene\_name )

Get gene information by gene name.

Parameters

<i>str</i>	Output, the gene's location.
<i>specie_name</i>	Input, the specie's name.
<i>gene_name</i>	Input, the gene's name.

Returns

0 for succeed, -1 for failed.

Definition at line 14 of file gene.cpp.

**4.4.2.4 cJSON\* getlineregion ( int *Chr\_No*, int *start*, int *end* )**

Get region information for part of a chromosome.

**Returns**

region information in json style.

Definition at line 59 of file region.cpp.

**4.4.2.5 int getRegion ( int *sgrna\_ID*, int *Chr\_No*, int *sgrna\_start*, int *sgrna\_end* )**

Get region information of an sgRNA.

**Returns**

region type of the sgRNA

**See also**

REGION\_EXON, REGION\_INTRON, REGION\_UTR, REGION\_INTERGENIC, REGION\_GENE

Definition at line 137 of file region.cpp.

**4.4.2.6 int localres\_count ( localrow \* *lr* )**

Get number of saved sgRNA-Info on RAM.

**See also**

[localrow](#)

Definition at line 47 of file localresult.cpp.

**4.4.2.7 int make\_mysqlres\_local ( localrow \*\* *localresult*, MYSQL\_RES \* *result\_t* )**

Save data from MYSQL\_RES to RAM.

**See also**

[localrow](#)

**Remarks**

Data saved in linked list.

Usage:

```
MYSQL_RES *result=mysql_store_result(mysql_conn);
```

```
localrow *localresult;
```

```
int res=make_mysqlres_local(&localresult,result);
```

Definition at line 16 of file localresult.cpp.

**4.5 region.cpp File Reference**

Get region information.

```
#include "main.h"
```

## Data Structures

- struct [node](#)  
*Local structure in getlineregion.*

## Typedefs

- typedef struct [node](#) [node](#)  
*Local structure in getlineregion.*

## Functions

- bool [cmp2](#) ([node](#) a, [node](#) b)  
*Compare to node by node.s, node.type.*
- int [get\\_Chr\\_No](#) (const char \*specie, const char \*chr\_name)  
*Get chromosome's Number(Chr\_No) by specie's name and chromosome's name.*
- int [region\\_wmin](#) (int a, int b)  
*Compare two type of region.*
- cJSON \* [getlineregion](#) (int Chr\_No, int start, int end)  
*Get region information for part of a chromosome.*
- int [getRegion](#) (int sgrna\_ID, int Chr\_No, int sgrna\_start, int sgrna\_end)  
*Get region information of an sgRNA.*

## Variables

- int [nt](#)  
*Local variables in getlineregion.*
- [node](#) [ns](#) [NODE\_SIZE]  
*Local variables in getlineregion.*
- int [region\\_str](#) [GENE\_LEN]  
*Local variables in getlineregion.*

### 4.5.1 Detailed Description

Get region information.

Author

Yi Zhao

Definition in file [region.cpp](#).

### 4.5.2 Function Documentation

#### 4.5.2.1 bool [cmp2](#) ( [node](#) a, [node](#) b )

Compare to node by node.s, node.type.

See also

[node](#)

Definition at line 17 of file [region.cpp](#).

**4.5.2.2** `int get_Chr_No ( const char * specie, const char * chr_name )`

Get chromosome's Number(Chr\_No) by specie's name and chromosome's name.

**Returns**

-1 means failed. Otherwise return Chr\_No.

Definition at line 26 of file region.cpp.

**4.5.2.3** `cJSON* getlineregion ( int Chr_No, int start, int end )`

Get region information for part of a chromosome.

**Returns**

region information in json style.

Definition at line 59 of file region.cpp.

**4.5.2.4** `int getRegion ( int sgrna_ID, int Chr_No, int sgrna_start, int sgrna_end )`

Get region information of an sgRNA.

**Returns**

region type of the sgRNA

**See also**

REGION\_EXON, REGION\_INTRON, REGION\_UTR, REGION\_INTERGENIC, REGION\_GENE

Definition at line 137 of file region.cpp.

**4.5.2.5** `int region_wmin ( int a, int b )`

Compare two type of region.

**Note**

priority: 0 1 2 3 4.

**See also**

REGION\_EXON, REGION\_INTRON, REGION\_UTR, REGION\_INTERGENIC, REGION\_GENE

Definition at line 46 of file region.cpp.

**4.6 score.cpp File Reference**

Include all functions about scoring and counting.

```
#include "main.h"
```

## Data Structures

- struct [thread\\_share\\_variables](#)  
*Share data between threads.*

## Functions

- bool [cmp](#) ([cJSON](#) \*a, [cJSON](#) \*b)  
*Compare two sgRNA's Smm in reverse order.*
- [cJSON](#) \* [cJSON\\_otj](#) ([localrow](#) \*lr, double oscore, int omms)  
*Package possible-offtarget sgRNA's Infomation to [cJSON](#) Object.*
- double [subscore](#) (int ini, [localrow](#) \*lr, int \*Nph, int type)  
*Calculate score of a possible-offtarget sgRNA (Smm).*
- void [score](#) ([localrow](#) \*lr, [localrow](#) row, int ini, int type, double r1)  
*Calculate score of a candidate sgRNA.*
- void \* [new\\_thread](#) (void \*args)  
*New thread function.*
- void [create\\_thread\\_socre](#) ([localrow](#) \*lr, [localrow](#) row, int ini, int type, double r1)  
*Function to create new thread.*

## Variables

- const double [M](#) [] = {0,0,0.014,0,0,0.395,0.317,0,0.389,0.079,0.445,0.508,0.613,0.851,0.732,0.828,0.615,0.804,0.685,0.583}  
*Weight Values. Reference: Genetic Screens in Human Cells Using the CRISPR-Cas9 System, Wang et al., 2014.*
- const double [eM](#) [] = {2.718281828,2.718281828,2.680491036,2.718281828,2.718281828,1.831252209,1.979808257,2.718281828,1.842272751,2.511800935,1.741940985,1.635584119,1.472556491,1.160672989,1.30734714,1.187677833,1.469614321,1.216526905,1.370259311,1.517402513}  
*Weight Values  $e^{(1-M[n])}$ .*
- struct [thread\\_share\\_variables](#) [thread\\_share\\_variables](#)

### 4.6.1 Detailed Description

Include all functions about scoring and counting.

#### Author

Yi Zhao

Definition in file [score.cpp](#).

### 4.6.2 Function Documentation

#### 4.6.2.1 [cJSON](#)\* [cJSON\\_otj](#) ( [localrow](#) \* lr, double oscore, int omms )

Package possible-offtarget sgRNA's Infomation to [cJSON](#) Object.

#### Parameters

<i>lr</i>	possible-offtarget sgRNA's infomation saved in structure localrow.
<i>oscore</i>	sgRNA's score(Smm).
<i>omms</i>	Number of mismatches.

See also

[localrow](#)

Definition at line 27 of file score.cpp.

**4.6.2.2 void score ( localrow \* lr, localrow row, int ini, int type, double r1 )**

Calculate score of a candidate sgRNA.

Parameters

<i>lr</i>	root pointer of saved possible-offtarget sgRNA in localrow linked list.
<i>row</i>	node of candidate sgRNA's infomation.
<i>ini</i>	ID of the candidate sgRNA.
<i>type</i>	Calculate type.
<i>r1</i>	Weight of Sspe.

See also

[localrow](#), [site](#), [in\\_site](#)

Definition at line 90 of file score.cpp.

**4.6.2.3 double subscore ( int ini, localrow \* lr, int \* Nph, int type )**

Calculate score of a possible-offtarget sgRNA (Smm).

Parameters

<i>ini</i>	ID of the candidate sgRNA.
<i>lr</i>	possible-offtarget sgRNA's infomation saved in structure localrow.
<i>Nph</i>	Nph(type=1) or nmm(type=0).
<i>type</i>	Nph's type.

See also

[localrow](#), [site](#), [in\\_site](#)

Definition at line 51 of file score.cpp.

## 4.7 util.cpp File Reference

Nothing here now.

### 4.7.1 Detailed Description

Nothing here now.

Definition in file [util.cpp](#).

## 4.8 util.h File Reference

Macros for compiling in both linux and windows.

```
#include <pthread.h>
#include <semaphore.h>
```

### Macros

- `#define mos_pthread_mutex_init pthread_mutex_init`
- `#define mos_sem_init sem_init`
- `#define mos_pthread_mutex_lock pthread_mutex_lock`
- `#define mos_pthread_mutex_unlock pthread_mutex_unlock`
- `#define mos_sem_wait sem_wait`
- `#define mos_sem_post sem_post`
- `#define mos_pthread_join pthread_join`
- `#define mos_pthread_detach pthread_detach`
- `#define mos_pthread_self pthread_self`
- `#define mos_pthread_create pthread_create`

### Typedefs

- `typedef pthread_mutex_t mos_pthread_mutex_t`
- `typedef sem_t mos_sem_t`
- `typedef pthread_t mos_pthread_t`
- `typedef pthread_attr_t mos_pthread_attr_t`

#### 4.8.1 Detailed Description

Macros for compiling in both linux and windows.

##### Author

Yi Zhao

##### Note

Document are gathered in PREDEFINED \_\_linux

Definition in file [util.h](#).